

Docket No.: 67471-022

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	:	Customer Number: 20277
Shohei MICHIMOTO, et al.	:	Confirmation Number: not yet assigned
Serial No.: not yet assigned	:	Group Art Unit: not yet assigned
Filed: August 01, 2003	:	Examiner: Not yet assigned
For:	:	COMPILER APPARATUS AND METHOD FOR DETERMINING LOCATIONS FOR DATA IN MEMORY AREA

**CLAIM OF PRIORITY AND
TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT**

Mail Stop
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 35 U.S.C. 119, Applicant(s) hereby claims(s) the priority of:

Japanese Patent Application 2002-225286, Filed August 1, 2002

Certified copy is submitted herewith.

Respectfully submitted,

MCDERMOTT, WILL & EMERY


Michael E. Fogarty
Registration No. 36,139

600 13th Street, N.W.
Washington, DC 20005-3096
(202) 756-8000 MEF:jgh
Facsimile: (202) 756-8087
Date: August 1, 2003

【代理人】

【識別番号】 100090446

【弁理士】

【氏名又は名称】 中島 司朗

【手数料の表示】

【予納台帳番号】 014823

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9003742

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データの記憶領域への配置を決定するコンパイラ装置及び
配置決定方法

【特許請求の範囲】

【請求項 1】 データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置であって、

複数のデータ属性を有する複数のデータの集合 X から、前記複数のデータ属性の何れかに基づいて定めた第 1 基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択手段と、

データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定手段と、

当該データの配置を前記記憶領域内に決定できない場合毎に、

選択されたデータの中から除外対象となるデータを、前記第 1 基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第 2 基準に従って特定する除外データ特定手段と

を備え、

前記配置決定データ選択手段は、前記データ集合 X から前記除外対象データを除いたデータの中から、前記第 1 基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合 X の全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返す

ことを特徴とするコンパイラ装置。

【請求項 2】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択手段は、アラインメントの降順に、前記集合 X から配置を決定すべきデータを選択し、

前記第 2 基準は、データのサイズの降順であり、

前記除外データ特定手段は、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項1記載のコンパイラ装置。

【請求項3】 前記コンパイラはさらに、前記配置決定データ選択手段による前記処理の繰返しが終了後、前記除外データ特定手段によって特定された除外対象データの中からデータのサイズの昇順に従って、順次データを選択する再配置決定データ選択手段を備え、

前記配置決定手段はさらに、前記再配置決定データ選択手段によってデータが選択される毎に当該データの前記記憶領域内への配置を決定する

ことを特徴とする請求項2記載のコンパイラ装置。

【請求項4】 前記配置決定データ選択手段はさらに、前記処理の繰返しを終了した後、前記除外データ特定手段によって特定された除外対象データの中から前記第1基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、

前記配置決定手段はさらに、前記配置決定データ選択手段によってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、

前記除外データ特定手段は、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第2基準に従って特定する再除外対象特定処理を行い、

前記配置決定データ選択手段は、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、

前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、

前記配置決定データ選択手段はさらに、再除外対象データについて、前記再選択処理を繰返し、

前記配置決定手段はさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、

前記除外データ特定手段はさらに、前記再配置決定処理により当該再除外対象

データの配置を決定できない場合毎に、前記再除外対象特定処理を繰返すことを特徴とする請求項 1 記載のコンパイラ装置。

【請求項 5】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データのサイズの降順である

ことを特徴とする請求項 4 記載のコンパイラ装置。

【請求項 6】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データの参照頻度の昇順である

ことを特徴とする請求項 4 記載のコンパイラ装置。

【請求項 7】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択手段は、アラインメントの降順に、前記集合 X から配置を決定すべきデータを選択し、

前記第 2 基準は、データの参照頻度の昇順であり、

前記除外データ特定手段は、データの参照頻度の昇順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項 1 記載のコンパイラ装置。

【請求項 8】 前記コンパイラはさらに、前記配置決定データ選択手段による前記処理の繰返しが終了後、特定された除外対象データの中からデータの参照頻度の降順に従って、順次データを選択する再配置決定データ選択手段を備え、

前記配置決定手段はさらに、前記再配置決定データ選択手段によってデータが選択される毎に当該データの前記記憶領域内への配置を決定する

ことを特徴とする請求項 7 記載のコンパイラ装置。

【請求項 9】 データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置におけるデータ配置方法であって、

複数のデータ属性を有する複数のデータの集合 X から、データ属性に基づいて定めた第 1 基準に従って、順次配置を決定すべきデータを選択する配置決定デ

タ選択ステップと、

データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと

当該データの配置を前記記憶領域内に決定できない場合に、

選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第2基準に従って特定する除外データ特定ステップと

を含み、

前記配置決定データ選択ステップは、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返す

ことを特徴とするデータ配置方法。

【請求項10】 前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、

前記第2基準は、データのサイズの降順であり、

前記除外データ特定ステップは、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項9記載のデータ配置決定方法。

【請求項11】 前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、

前記第2基準は、データの参照頻度の昇順であり、

前記除外データ特定ステップは、データの参照頻度の昇順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項9記載のデータ配置決定方法。

【請求項 1 2】 前記配置決定データ選択ステップはさらに、前記処理の繰返しを終了した後、前記除外データ特定ステップによって特定された除外対象データの中から前記第 1 基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、

前記配置決定ステップはさらに、前記配置決定データ選択ステップによってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、

前記除外データ特定ステップは、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第 2 基準に従って特定する再除外対象特定処理を行い、

前記配置決定データ選択ステップは、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、

前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、

前記配置決定データ選択ステップはさらに、再除外対象データについて、前記再選択処理を繰返し、

前記配置決定ステップはさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、

前記除外データ特定ステップはさらに、前記再配置決定処理により当該再除外対象データの配置を決定できない場合毎に、前記再除外対象特定処理を繰返す

ことを特徴とする請求項 9 記載のデータ配置決定方法。

【請求項 1 3】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データのサイズの降順である

ことを特徴とする請求項 1 2 記載のデータ配置決定方法。

【請求項 1 4】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第2基準は、データの参照頻度の昇順である

ことを特徴とする請求項12記載のデータ配置決定方法。

【請求項15】 データの記憶領域内における位置を特定するための即値をmビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置に用いられるプログラムであって、

前記プログラムは、

複数のデータ属性を有する複数のデータの集合Xから、データ属性に基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、

データが選択される毎に、選択されたデータのn ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと

当該データの配置を前記記憶領域内に決定できない場合に、

選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第2基準に従って特定する除外データ特定ステップと

を含み、

前記配置決定データ選択ステップは、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返す

ことを特徴とするプログラム。

【請求項16】 前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、

前記第2基準は、データのサイズの降順であり、

前記除外データ特定ステップは、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項15記載のプログラム。

【請求項 1 7】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択ステップは、アラインメントの降順に、前記集合 X から配置を決定すべきデータを選択し、

前記第 2 基準は、データの参照頻度の昇順であり、

前記除外データ特定ステップは、データの参照頻度の昇順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項 1 5 記載のプログラム。

【請求項 1 8】 前記配置決定データ選択ステップはさらに、前記処理の繰返しを終了した後、前記除外データ特定ステップによって特定された除外対象データの中から前記第 1 基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、

前記配置決定ステップはさらに、前記配置決定データ選択ステップによってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、

前記除外データ特定ステップは、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第 2 基準に従って特定する再除外対象特定処理を行い、

前記配置決定データ選択ステップは、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、

前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、

前記配置決定データ選択ステップはさらに、再除外対象データについて、前記再選択処理を繰返し、

前記配置決定ステップはさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、

前記除外データ特定ステップはさらに、前記配置決定処理により当該再除外対

象データの配置を決定できない場合毎に、再除外対象特定処理を繰返すことを特徴とする請求項 1 5 記載のプログラム。

【請求項 1 9】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データのサイズの降順である

ことを特徴とする請求項 1 8 記載のプログラム。

【請求項 2 0】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データの参照頻度の昇順である

ことを特徴とする請求項 1 8 記載のプログラム。

【請求項 2 1】 データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラに用いられるプログラムを記録したコンピュータ読取可能な記録媒体であって、

前記プログラムは、

複数のデータ属性を有する複数のデータの集合 X から、データ属性に基づいて定めた第 1 基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、

データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと

当該データの配置を前記記憶領域内に決定できない場合に、

選択されたデータの中から除外対象となるデータを、前記第 1 基準とは異なる、データ属性に基づいて定めた第 2 基準に従って特定する除外データ特定ステップと

を含み、

前記配置決定データ選択ステップは、前記データ集合 X から前記除外対象データを除いたデータの中から、前記第 1 基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合 X の全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返す

ことを特徴とするコンピュータ読取可能な記録媒体。

【請求項 2 2】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択ステップは、アラインメントの降順に、前記集合 X から配置を決定すべきデータを選択し、

前記第 2 基準は、データのサイズの降順であり、

前記除外データ特定ステップは、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定する

ことを特徴とする請求項 2 1 記載のコンピュータ読取可能な記録媒体。

【請求項 2 3】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記配置決定データ選択ステップは、アラインメントの降順に、前記集合 X から配置を決定すべきデータを選択し、

前記第 2 基準は、データの参照頻度の昇順であり、

前記除外データ特定ステップは、データの参照頻度の昇順に、選択されたデータの集合から除外対象となるデータを特定する

ことを特徴とする請求項 2 1 記載のコンピュータ読取可能な記録媒体。

【請求項 2 4】 前記配置決定データ選択ステップはさらに、前記処理の繰返しを終了した後、前記除外データ特定ステップによって特定された除外対象データの中から前記第 1 基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、

前記配置決定ステップはさらに、前記配置決定データ選択ステップによってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、

前記除外データ特定ステップは、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第 2 基準に従って特定する再除外対象特定処理を行い、

前記配置決定データ選択ステップは、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入る

ように決定されるまで繰返し、

前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、

前記配置決定データ選択ステップはさらに、再除外対象データについて、前記再選択処理を繰返し、

前記配置決定ステップはさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、

前記除外データ特定ステップはさらに、前記配置決定処理により当該再除外対象データの配置を決定できない場合毎に、再除外対象特定処理を繰返す

ことを特徴とする請求項 2 1 記載のコンピュータ読取可能な記録媒体。

【請求項 2 5】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データのサイズの降順である

ことを特徴とする請求項 2 4 記載のコンピュータ読取可能な記録媒体。

【請求項 2 6】 前記第 1 基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、

前記第 2 基準は、データの参照頻度の昇順である

ことを特徴とする請求項 2 4 記載のコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、ソースプログラムを機械語コードのオブジェクトプログラムに翻訳するコンパイラに関し、特にデータの記憶領域への配置を決定するコンパイラ装置及びコンパイラ装置による変数データの記憶領域への配置決定方法に関する。

【0 0 0 2】

【従来の技術】

近年、ソフトウェアの大規模化に伴い、大規模システムの開発に適した C 言語や C++ 言語などの高級言語がソフトウェアに使用されるようになってきている。これらの高級言語で記述されたソースプログラムは、専用のコンパイラによって

CPUが実行可能な機械語コードに翻訳された後、メモリ上にロードされ、実行される。メモリ上にロードされるプログラムは、命令を示すコードと変数を示すデータとの組から構成され、コードの示す命令は、当該コードと組のデータを参照することにより実行される。

【0003】

ここで、データには、プログラムの実行中常にデータを格納するための記憶領域が確保される大域変数と所定の関数が呼び出されたときにデータを格納するための記憶領域（スタック領域）が確保される自動変数の2種類がある。以下、「自動変数」を「変数」と呼ぶ。

コンパイラは、所定の各関数と組の変数に対して、スタック領域における配置を決定する。図12は、従来法1における、コンパイラによるスタック領域における変数の配置の決定処理を示すフローチャートである。

【0004】

コンパイラは、ソースプログラムを読み出し（ステップS1）、読み出したプログラムの構文解析の処理を実行し（ステップS2）、解析した結果に基づいて、スタック領域への配置対象となる変数対照表を作成する（ステップS3）。

ここで、「変数対照表」とは、変数の型と変数名とデータサイズ（以下、「サイズ」という。）とアラインメントとの対応関係を示す表のことをいう。

【0005】

又、「アラインメント」とは、変数の型に応じて予め定められている、当該変数のスタック領域への配置に対する制約の大きさを示す値のことであり、例えば、アラインメントが2の場合は、当該変数をアドレスが2の倍数のスタック領域に配置しなければならないことを示し、アラインメントが4の場合は、アドレスが4の倍数のスタック領域に配置しなければならないことを示す。

【0006】

図4は、変数対照表の具体例を示す図である。図4の例では、4つの変数の型（char,int,char[],double[]）について、上記の対応関係を示している。

コンパイラは、変数対照表の各変数をスタック領域への配置対象集合の構成要素とし（ステップS4）、配置対象集合からサイズが最小の変数を選択し、当該

変数のスタック領域における配置を決定し（ステップS5）、配置対象集合の全ての変数について配置を決定したか否かを判定する（ステップS6）。

【0007】

全ての変数について配置を決定した場合（ステップS6：Y）は、処理を終了し、決定していない場合（ステップS6：N）、ステップS5の処理に戻る。

例えば、図4の変数対象表に示す変数の集合を配置対象集合とした場合には、コンパイラによって決定される変数のスタック領域における配置は、図13に示す配置模式図ようになる。図13は、各変数がサイズの小さい順（a,b,c,dの順）にスタック領域に配置されていることを示している。

【0008】

ここで、図13の各マス目で示す部分には、1バイトの変数が格納され、アドレスは、先頭のマス目を0番地、その右隣のマス目を1番地、そのまた右隣のマス目を2番地、先頭のマス目の下のマス目を8番地というように付与されているものとする。以下、後述する配置模式図についても同様とする。

図13のスタック領域の空き領域の部分は、アラインメントの制約により、変数を配置できなかったスタック領域の空き領域を示している。例えば、変数bのアラインメントは4であるので、コンパイラはbをaの隣のアドレスである1番地から配置することができず、4番地のアドレスから配置している。同様に、変数dは、16番地のアドレスから配置している。

【0009】

これにより、例えば、スタック領域に格納された変数に1命令でアクセス可能な範囲を32バイトであるとする、1命令で3つの型の異なる変数にアクセスすることが可能となる。

このように、サイズの小さい順にスタック領域における変数の配置を決定することにより、サイズの小さい変数の呼出頻度が高いプログラムにおいて、プログラムの実行速度を速めることができる。

【0010】

又、従来法2として、変数のアラインメントの大きい順に、スタック領域における変数の配置の決定する方法がある。

図14は、アラインメントが大きい順に従ってコンパイラが行うスタック領域における変数の配置の決定処理を示すフローチャートである。

コンパイラは、ソースプログラムを読み出し（ステップS11）、読み出したプログラムの構文解析の処理を実行し（ステップS12）、解析した結果に基づいて、スタック領域への配置対象となる変数対照表を作成し（ステップS13）、変数対照表の各変数をスタック領域への配置対象集合の構成要素とし（ステップS14）、配置対象集合からアラインメントが最大の変数を選択し、当該変数のスタック領域における配置を決定し（ステップS15）、配置対象集合の全ての変数について配置を決定したか否かを判定する（ステップS16）。

【0011】

全ての変数について配置を決定した場合（ステップS16：Y）は、処理を終了し、決定していない場合（ステップS16：N）、ステップS15の処理に戻る。

例えば、図4の変数対照表に示す変数の集合を配置対象データ集合とした場合には、コンパイラによって決定される変数のスタック領域における配置は、図15に示す配置模式図のようになる。図15は、各変数がアラインメントの大きい順（d,b,c,a（cとaの順は逆でもよい）の順）にスタック領域に配置されていることを示している。

【0012】

図15に示すように、この方法によると、コンパイラは、スタック領域に空き領域を生じさせることなく、各変数をスタック領域に配置することが可能となる。

このように、アラインメントの大きい順にスタック領域における変数の配置を決定することにより、メモリの無駄な使用をなくし、少ないメモリ容量で効率よく、変数をスタック領域に格納することができる。

【0013】

【発明が解決しようとする課題】

しかしながら、従来法1によって、スタック領域における変数の配置を決定した場合には、スタック領域に無駄な空き領域が生じ、メモリ使用量が大きくなる

という不具合が生じる。

又、従来法2によって、スタック領域における変数の配置を決定した場合には、サイズの大きい変数がスタック領域の先頭に配置されることがあり、1命令でアクセス可能な変数の数が少なくなり、アラインメントが小さい変数の呼出頻度が多いプログラムにおいて、プログラムの実行速度を低下させるという不具合が生じる。

【0014】

そこで、本発明は、スタック領域における変数の配置の決定を最適化するコンパイラ装置及び配置決定方法を提供することを目的とする。

【0015】

【課題を解決するための手段】

上記課題を解決するために本発明は、データの記憶領域内における位置を特定するための即値をmビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置であって、複数のデータ属性を有する複数のデータの集合Xから、前記複数のデータ属性の何れかに基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択手段と、データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定手段と、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第2基準に従って特定する除外データ特定手段とを備え、前記配置決定データ選択手段は、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すことを特徴とする。

【0016】

又、本発明は、データの記憶領域内における位置を特定するための即値をmビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置におけるデータ配置方法であって、複数のデータ属性を有する複数のデータの集合

Xから、データ属性に基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと、当該データの配置を前記記憶領域内に決定できない場合に、選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第2基準に従って特定する除外データ特定ステップとを含み、前記配置決定データ選択ステップは、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すことを特徴とする。

【0017】

又、本発明は、データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置に用いられるプログラムであって、前記プログラムは、複数のデータ属性を有する複数のデータの集合Xから、データ属性に基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと、当該データの配置を前記記憶領域内に決定できない場合に、選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第2基準に従って特定する除外データ特定ステップとを含み、前記配置決定データ選択ステップは、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すことを特徴とする。

【0018】

又、本発明は、データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラに用い

られるプログラムを記録したコンピュータ読取可能な記録媒体であって、前記プログラムは、複数のデータ属性を有する複数のデータの集合Xから、データ属性に基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、データが選択される毎に、選択されたデータのn ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと当該データの配置を前記記憶領域内に決定できない場合に、選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、データ属性に基づいて定めた第2基準に従って特定する除外データ特定ステップとを含み、前記配置決定データ選択ステップは、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すことを特徴とする。

【0019】

【発明の実施の形態】

<構成>

図1は、本実施の形態におけるコンパイラ装置100の構成を示す。コンパイラ装置100は、CPU101、表示部102、入力部103、記憶部104、から構成される。

【0020】

CPU101は、入力部103から入力されるユーザー指示に応じて、記憶部104に格納されているコンパイラプログラム204を実行することにより、コンパイルを行い、ソースプログラム205に含まれる変数のスタック領域における配置を決定し、ソースプログラム205のコードをオブジェクトプログラムのコード305に翻訳する。

【0021】

具体的には、記憶部104からソースプログラム205を読み出し、読み出したソースプログラム205の構文解析の処理を実行し、解析した結果に基づいて、スタック領域への変数の配置決定処理を行い、オブジェクトプログラム305のコードを生成する。

なお、配置決定処理の詳細については後述する。

【0022】

表示部102は、コンパイル処理の結果を表示する。

入力部103は、ユーザーからコンパイルの実行指示の入力を受付ける。

記憶部104は、コンパイラプログラム204とソースプログラム205を記憶している。又、コンパイルの実行によりソースプログラムを翻訳したオブジェクトプログラム305を記憶する。

【0023】

図2は、ソースプログラム205の具体例を示す。図2には、ソースプログラム205の一部の例が示されている。図2の102~200は、関数f(void)を定義している部分であり、110~113は当該関数の変数を宣言している部分である。この宣言部分の記述に基いて、図4に示すような変数対照表がCPU101の実行するコンパイル処理によって作成される。

<動作>

次にCPU101の行うスタック領域における変数の配置決定処理について説明する。図3は、上記配置決定処理を示すフローチャートである。以下、図3に示すフローチャートを参照して説明する。

【0024】

CPU101は、記憶部104からソースプログラム205を読み出し、読み出したプログラムの構文解析の処理を実行し、解析した結果に基いて、変数対照表を作成し（ステップS1002）、除外集合を空集合とし（ステップS1003）、変数対照表に含まれる各変数から構成される配置対象集合を作成し（ステップS1004）、配置対象集合からアラインメントが最大の変数を選択し（ステップS1005）、当該変数のアラインメントに基いて、スタック領域の空き領域内の、当該変数を配置可能で配置がまだ未決定の領域において、当該変数のアドレスが最小となるように当該変数の配置を決定し（ステップS1006）、配置を決定した変数を配置対象集合から除き（ステップS1007）、配置を決定した変数を配置決定済集合の構成要素とし（ステップS1008）、決定した配置が、スタック領域における先頭アドレスが示す領域から所定の範囲内の領域に

あるか否かを判定する（ステップ S 1 0 0 9）。

【 0 0 2 5 】

所定の範囲内の領域に無い場合（ステップ S 1 0 0 9 : N）、配置決定済集合からサイズが最大の変数（集合に含まれる変数が 1 つの場合は、当該変数）を特定し、特定した変数を配置決定済集合から除き、除外集合に加え（ステップ S 1 0 1 0）、特定した変数を除く残りの配置決定済集合に含まれる各変数を配置対象集合に加え（ステップ S 1 0 1 1）、ステップ S 1 0 0 5～ステップ S 1 0 0 9 の処理を繰り返す。

【 0 0 2 6 】

所定の範囲内の領域に有る場合（ステップ S 1 0 0 9 : Y）、配置対象集合に含まれる変数が無くなったか否かを判定する（ステップ S 1 0 1 2）。

無くなっていない場合（ステップ S 1 0 1 2 : N）、ステップ S 1 0 0 5 の処理に戻る。

無くなっている場合（ステップ S 1 0 1 2 : Y）、除外集合に含まれる変数があるか否かを判定する（ステップ S 1 0 1 3）。

【 0 0 2 7 】

変数がある場合（ステップ S 1 0 1 3 : Y）、除外集合からサイズが最小の変数（除外集合に含まれる変数が 1 個の場合は、当該変数）を選択し（ステップ S 1 0 1 4）、当該変数のアラインメント値に基いて、スタック領域の空き領域内の、当該変数を配置可能で、配置がまだ未決定の領域において、当該変数のアドレスが最小となるように当該変数の配置を決定し（ステップ S 1 0 1 5）、配置を決定した当該変数を除外集合から除き（ステップ S 1 0 1 6）、ステップ S 1 0 1 3 の処理に戻る。

【 0 0 2 8 】

変数がない場合（ステップ S 1 0 1 3 : N）、処理を終了する。

具体例として図 2 の例を用いて説明すると、CPU 1 0 1 は、主記憶部 1 0 4 から、図 2 に示すソースプログラムの一部を含むソースプログラムを読み出し、読み出したプログラムの構文解析の処理を実行し、解析した結果に基いて、図 4 に示す変数の変数対照表を作成し（ステップ S 1 0 0 2）、当該変数対照表に含ま

れる各変数であるa,b,c,dから構成される配置対象集合を作成し（ステップS1004）、当該配置対象集合からアラインメントが最大の変数であるdを取り出し（ステップS1005）、変数dのアラインメント値が8であるので、スタック領域の空き領域内の、変数dを配置可能で配置がまだ未決定の領域（アドレスが8の倍数である領域）において、変数dを配置するアドレス（ここでは、スタック領域の各1バイト領域毎に、先頭から順に0番地、1番地、2番地というようにアドレスが付与されるものとする。）が最小となる0番地から変数dを配置することを決定し（ステップS1006）、変数dを配置対象集合から除き（ステップS1007）、変数dを配置決定済集合の構成要素とし（ステップS1008）、決定した配置が、スタック領域における先頭アドレスが示す領域から所定の範囲内の領域（ここでは、先頭アドレス0番地から31番地の示す32バイトの領域を所定の範囲の領域とする。）にあるか否かを判定する（ステップS1009）。この段階における変数dのスタック領域における配置を示す配置模式図を図5に示す。

【0029】

変数dの配置は、アドレス0番地からであるので、CPU101は、所定の範囲内の領域に有ると判定し（ステップS1009：Y）、配置対象集合に含まれる変数が無くなったか否かを判定する（ステップS1012）。

配置対象集合には、a,b,cの3つの変数が残っているので、CPU101は、無くなっていないと判定し（ステップS1012：N）、ステップS1005において、アラインメントが最大である変数bを配置対象集合から選択し、変数bのアラインメントが4であるので、0番地から31番地のスタック領域には、図5に示すように既に変数dが配置されているので、変数bを配置可能で配置が未決定の領域（アドレスが4の倍数である領域で、アドレスが32番地以降のスタック領域）において、変数bを配置するアドレスが最小となる32番地から変数bを配置することを決定し（ステップS1006）、変数bを配置対象集合から除き（ステップS1007）、変数bを配置決定済集合の構成要素とし（ステップS1008）、決定した配置が、スタック領域における先頭アドレス0番地から31番地の示す領域の範囲内にあるか否かを判定する（ステップS1009）。こ

の段階における変数d,bのスタック領域における配置を示す配置模式図を図6に示す。

【0030】

変数bの配置は、アドレス32番地からであるので、CPU101は、所定の範囲内の領域に無いと判定し（ステップS1009：N）、変数bと変数dとから成る配置決定済集合からサイズが最大の変数dを特定し、特定した変数dを配置決定済集合から除き、除外集合に加え（ステップS1010）、特定した変数dを除く残りの配置決定済集合に含まれる変数bを配置対象集合に加え（ステップS1011）、ステップS1005～ステップS1009及びステップS1012の処理を繰り返し、変数b,c,aの順（cとaの順序は逆であってもよい。）に配置を決定し、変数b,c,aを配置決定済集合の構成要素とする。

【0031】

これにより、変数b,c,aは、図7の配置模式図に示すようにスタック領域のアドレスが0番地から23番地のスタック領域に配置されるので、CPU101は、所定の範囲内の領域に有ると判定し（ステップS1009：Y）、さらに配置対象集合に含まれる変数が無くなったと判定し（ステップS1012：Y）、除外集合に変数dが加えられているので、ステップS1013において、変数があると判定し（ステップS1013：Y）、除外集合からサイズが最小の変数d（この例では、除外集合には変数dのみが含まれているので、変数dが選択される。）を選択し（ステップS1014）、当該変数のアラインメントが8であるので、スタック領域の空き領域内の当該変数を配置可能な領域（24番地以降の領域でアドレスが8の倍数の領域）において、変数dのアドレスが最小となる24番地から変数dを配置することを決定し（ステップS1015）、変数dを除外集合から除き（ステップS1016）、さらに除外集合に含まれる変数が無いと判定し（ステップS1013：N）、配置決定処理を終了する。

【0032】

これにより、変数a,b,c,dは、図8の配置模式図に示されるように配置されることになり、スタック領域に格納された変数に1命令でアクセス可能な範囲を例えば、32バイトであるとする、上記の例においては、1命令でソースプログ

ラムに記述されている全ての変数にアクセスすることが可能となる。

図 9 は、上記の配置決定処理によって決定されたスタック領域の配置に変数 a, b, c, d が配置された場合におけるオブジェクトプログラム例を示す。図 9 の 2 1 0 は、スタック領域の 2 3 番地に格納されている変数 a をレジスタ r0 に格納する命令を示している。同様に、図 9 の 2 2 0 は、スタック領域の 0 番地に格納されている変数 b をレジスタ r0 に格納する命令を、図 9 の 2 3 0 は、スタック領域の 4 番地に格納されている変数 c をレジスタ r0 に格納する命令を、図 9 の 2 4 0 は、スタック領域の 2 4 番地に格納されている変数 d をレジスタ r0 に格納する命令をそれぞれ示している。

【0033】

一方、図 1 0 は、従来法 1 によって、図 1 1 は、従来法 2 よって、それぞれ変数 a, b, c, d がスタック領域に配置された場合におけるオブジェクトプログラム例を示す。

図 1 0 では、スタック領域の 0 番地、4 番地、8 番地にそれぞれ格納されている変数 a, b, c をそれぞれレジスタ r0 に格納する命令 3 1 0 ~ 3 3 0 は、1 命令になっているが、図 1 0 の 3 4 0 及び 3 5 0 に示すように、スタック領域の 3 2 番地に格納されている変数 d をレジスタ r0 に格納する命令が 2 命令になっており、命令コードのサイズも増加している。

【0034】

同様に図 1 1 では、スタック領域の 0 番地に格納されている変数 d をレジスタ r0 に格納する命令 4 7 0 は 1 命令になっているが、図 1 1 の 4 1 0 及び 4 2 0、4 3 0 及び 4 4 0、4 5 0 及び 4 6 0 に示すように、スタック領域に格納されている変数 a, b, c のそれぞれをレジスタ r0 に格納する各命令が 2 命令になっており、命令コードのサイズも増加している。

【0035】

このように、本実施の形態における配置決定処理によって変数の配置を実行することにより、スタック領域の変数へのアクセスに要する命令数や命令のコードサイズを少なくすることができる。

<補足>

以上、本発明に係るコンパイラ装置について、実施の形態に基づいて説明したが、本発明はこれら実施の形態に限られないことは勿論である。即ち、

(1) 本実施の形態においては、図3のステップS1005において、アライメントを基準として、配置対象集合から変数を選択することとしたが、他のデータ属性を基準としてもよい。例えば、図16に示すように、変数対照表に各変数の参照頻度に関する情報を加え、参照頻度の多い順に変数を選択することとしてもよい。又、スタック領域に格納されている変数にアクセスするために、制約の少ない命令を使用可能か否か、或いはコードサイズの小さい命令を使用可能か否かを基準としてもよい。

【0036】

同様に図3のステップS1010において、サイズを基準として、配置決定済集合から除外する変数を特定することとしたが、他のデータ属性を基準としてもよい。例えば、図16に示すように、変数対照表に各変数の参照頻度に関する情報を加え、参照頻度の少ない順に変数を特定することとしてもよい。参照頻度に関する情報は、例えばソースプログラムの構文解析の結果に基づいて作成したり、シュミレータを用いてソースプログラムを実行させた結果に基づいて作成することができる。

【0037】

又、スタック領域に格納されている変数にアクセスするために、制約の少ない命令を使用可能か否か、或いはコードサイズの小さい命令を使用可能か否かを基準としてもよい。

(2) 同様に図3のステップ1013において、変数のデータサイズを基準にして、除外集合から変数を選択することとしたが、他のデータ属性を基準としてもよい。例えば、図16に示すように、変数対照表に各変数の参照頻度に関する情報を加え、参照頻度の多い順に変数を選択することとしてもよい。

【0038】

又、スタック領域に格納されている変数にアクセスするために、制約の少ない命令を使用可能か否か、或いはコードサイズの小さい命令を使用可能か否かを基準としてもよい。

(3) 図3のフローチャートに示すスタック領域における変数の配置決定処理においては、除外集合に含まれる変数の配置決定をステップS1013～ステップS1016に示す処理によって行うこととしたが、上記処理の代わりに除外集合に含まれる変数に対して、ステップS1005～ステップS1012に示す処理を繰返すことにより、所定の範囲のスタック領域毎に当該変数の配置決定を行うこととしてもよい。

【0039】

例えば、上記処理によって、0番地から31番地までの32バイトのスタック領域内において最初に配置対象集合に含まれる各変数の配置を決定し、次に当該スタック領域内において配置を決定できなかった除外集合に含まれる変数に対して、32番地から63番地までの32バイトのスタック領域内を割付対象領域として、各変数の配置を決定し、さらに64番地から95番地までの32バイトのスタック領域内を割付対象領域として、同様の配置決定を繰返すというようにして、全ての変数に対して、配置が決定されるまで所定の範囲のスタック領域毎に配置決定を繰返すこととしてもよい。

【0040】

なお、除外集合に含まれる変数に対して配置決定を繰返す領域は、上記のように同一サイズの領域毎とは限らず、スタック領域のオフセット境界(命令サイズ、レイテンシ、命令の組み合わせ等により決定される)によって分けられ、そのオフセット値が小さい領域から割付対象領域となる。

【0041】

【発明の効果】

(1) 本発明は、データの記憶領域内における位置を特定するための即値をmビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置であって、複数のデータ属性を有する複数のデータの集合Xから、前記複数のデータ属性の何れかに基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択手段と、データが選択される毎に、選択されたデータのn ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定手段と当該データの配置を前記記憶領域内に決定できない場合毎に、選択され

たデータの中から除外対象となるデータを、前記第 1 基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第 2 基準に従って特定する除外データ特定手段とを備え、前記配置決定データ選択手段は、前記データ集合 X から前記除外対象データを除いたデータの中から、前記第 1 基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合 X の全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すことを特徴とする構成を備える。

【 0 0 4 2 】

(2) 又、前記配置決定データ選択手段はさらに、前記処理の繰返しを終了した後、前記除外データ特定手段によって特定された除外対象データの中から前記第 1 基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、前記配置決定手段はさらに、前記配置決定データ選択手段によってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、前記除外データ特定手段は、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第 2 基準に従って特定する再除外対象特定処理を行い、前記配置決定データ選択手段は、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、前記配置決定データ選択手段はさらに、再除外対象データについて、前記再選択処理を繰返し、前記配置決定手段はさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、前記除外データ特定手段はさらに、前記再配置決定処理により当該再除外対象データの配置を決定できない場合毎に、前記再除外対象特定処理を繰返すこととしてもよい。

【 0 0 4 3 】

(3) 又、本発明は、データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置におけるデータ配置方法であって、複数のデータ属性を有する複数のデータ

の集合Xから、データ属性に基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと当該データの配置を前記記憶領域内に決定できない場合に、選択されたデータの中から除外対象となるデータを、前記第1基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第2基準に従って特定する除外データ特定ステップとを含み、前記配置決定データ選択ステップは、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すこととしてもよい。

【0044】

(4) 又、前記配置決定データ選択ステップはさらに、前記処理の繰返しを終了した後、前記除外データ特定ステップによって特定された除外対象データの中から前記第1基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、前記配置決定ステップはさらに、前記配置決定データ選択ステップによってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、前記除外データ特定ステップは、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第2基準に従って特定する再除外対象特定処理を行い、前記配置決定データ選択ステップは、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、前記配置決定データ選択ステップはさらに、再除外対象データについて、前記再選択処理を繰返し、前記配置決定ステップはさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、前記除外データ特定ステップはさらに、前記再配置決定処理により当該再除外対象データの配置を決定できない場合毎に、前記再除外対象

特定処理を繰返すこととしてもよい。

【 0 0 4 5 】

(5) 又、本発明は、データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラ装置に用いられるプログラムであって、前記プログラムは、複数のデータ属性を有する複数のデータの集合 X から、データ属性に基づいて定めた第 1 基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと、当該データの配置を前記記憶領域内に決定できない場合に、選択されたデータの中から除外対象となるデータを、前記第 1 基準とは異なる、前記複数のデータ属性の何れかに基いて定めた第 2 基準に従って特定する除外データ特定ステップとを含み、前記配置決定データ選択ステップは、前記データ集合 X から前記除外対象データを除いたデータの中から、前記第 1 基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合 X の全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すこととしてもよい。

【 0 0 4 6 】

(6) 又、前記配置決定データ選択ステップはさらに、前記処理の繰返しを終了した後、前記除外データ特定ステップによって特定された除外対象データの中から前記第 1 基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、前記配置決定ステップはさらに、前記配置決定データ選択ステップによってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、前記除外データ特定ステップは、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対象となるデータを、前記第 2 基準に従って特定する再除外対象特定処理を行い、前記配置決定データ選択ステップは、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、前記再選択処理の繰返しを終了した後、配置未決定の

再除外対象データが残っている場合に限り、前記配置決定データ選択ステップはさらに、再除外対象データについて、前記再選択処理を繰返し、前記配置決定ステップはさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、前記除外データ特定ステップはさらに、前記配置決定処理により当該再除外対象データの配置を決定できない場合毎に、再除外対象特定処理を繰返すこととしてもよい。

【 0 0 4 7 】

(7) 又、本発明は、データの記憶領域内における位置を特定するための即値を m ビットで指定する命令を含む命令セットの命令コードを生成するコンパイラに用いられるプログラムを記録したコンピュータ読取可能な記録媒体であって、前記プログラムは、複数のデータ属性を有する複数のデータの集合 X から、データ属性に基づいて定めた第 1 基準に従って、順次配置を決定すべきデータを選択する配置決定データ選択ステップと、データが選択される毎に、選択されたデータの n ($n \leq 2^m$) バイトの記憶領域内への配置の決定を試みる配置決定ステップと当該データの配置を前記記憶領域内に決定できない場合に、選択されたデータの中から除外対象となるデータを、前記第 1 基準とは異なる、データ属性に基づいて定めた第 2 基準に従って特定する除外データ特定ステップとを含み、前記配置決定データ選択ステップは、前記データ集合 X から前記除外対象データを除いたデータの中から、前記第 1 基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合 X の全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すこととしてもよい。

【 0 0 4 8 】

(8) 又、前記配置決定データ選択ステップはさらに、前記処理の繰返しを終了した後、前記除外データ特定ステップによって特定された除外対象データの中から前記第 1 基準に従って、順次配置を決定すべきデータを再選択する再選択処理を行い、前記配置決定ステップはさらに、前記配置決定データ選択ステップによってデータが再選択される毎に新たな記憶領域内への当該データの配置を決定する再配置決定処理を行い、前記除外データ特定ステップは、当該データの配置を前記記憶領域内に決定できない場合毎に、選択されたデータの中から再除外対

象となるデータを、前記第2基準に従って特定する再除外対象特定処理を行い、前記配置決定データ選択ステップは、前記除外対象データの中から前記再除外対象データを除いたデータ群について、前記再選択処理を、再除外対象データを除く前記除外対象データの全てのデータの配置が前記新たな記憶領域内に入るように決定されるまで繰返し、前記再選択処理の繰返しを終了した後、配置未決定の再除外対象データが残っている場合に限り、前記配置決定データ選択ステップはさらに、再除外対象データについて、前記再選択処理を繰返し、前記配置決定ステップはさらに、配置を決定すべき再除外対象データが選択される毎に前記再配置決定処理を繰返し、前記除外データ特定ステップはさらに、前記配置決定処理により当該再除外対象データの配置を決定できない場合毎に、再除外対象特定処理を繰返すこととしてもよい。

【0049】

これにより、所定サイズの記憶領域におけるデータの配置の決定を最適化することができる。

ここで、(1)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択手段は、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データのサイズの降順であり、前記除外データ特定手段は、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0050】

又、(1)の構成において、前記コンパイラはさらに、前記配置決定データ選択手段による前記処理の繰返しが終了後、前記除外データ特定手段によって特定された除外対象データの中からデータのサイズの昇順に従って、順次データを選択する再配置決定データ選択手段を備え、前記配置決定手段はさらに、前記再配置決定データ選択手段によってデータが選択される毎に当該データの前記記憶領域内への配置を決定することとしてもよい。

【0051】

又、(2)の構成において、前記第1基準は、データの前記記憶領域への配置

の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データのサイズの降順であることとしてもよい。

又、(3)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データのサイズの降順であり、前記除外データ特定ステップは、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0052】

又、(4)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データのサイズの降順であることとしてもよい。

又、(5)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データのサイズの降順であり、前記除外データ特定ステップは、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0053】

又、(6)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データのサイズの降順であることとしてもよい。

又、(7)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データのサイズの降順であり、前記除外データ特定ステップは、データのサイズの降順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0054】

又、(8)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データのサイズの降順であることとしてもよい。

これにより、所定サイズの記憶領域に無駄な空き領域を生じさせることなく、少ない命令数でできるだけ多くの種類のデータにアクセス可能となるように、データの記憶領域における配置を決定することができる。

【0055】

ここで、(2)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データの参照頻度の昇順であることとしてもよい。

又、(1)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択手段は、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データの参照頻度の昇順であり、前記除外データ特定手段は、データの参照頻度の昇順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0056】

又、前記コンパイラはさらに、前記配置決定データ選択手段による前記処理の繰返しが終了後、特定された除外対象データの中からデータの参照頻度の降順に従って、順次データを選択する再配置決定データ選択手段を備え、前記配置決定手段はさらに、前記再配置決定データ選択手段によってデータが選択される毎に当該データの前記記憶領域内への配置を決定することとしてもよい。

【0057】

又、(3)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データの参照頻度の昇順であり、前記除外データ特定ステップは、データの参照頻度の昇順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0058】

又、(4)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データの参照頻度の昇順であることとしてもよい。

又、(5)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データの参照頻度の昇順であり、前記除外データ特定ステップは、データの参照頻度の昇順に、選択されたデータの中から除外対象となるデータを特定することとしてもよい。

【0059】

又、(6)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データの参照頻度の昇順であることとしてもよい。

又、(7)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記配置決定データ選択ステップは、アラインメントの降順に、前記集合Xから配置を決定すべきデータを選択し、前記第2基準は、データの参照頻度の昇順であり、前記除外データ特定ステップは、データの参照頻度の昇順に、選択されたデータの集合から除外対象となるデータを特定することとしてもよい。

【0060】

又、(8)の構成において、前記第1基準は、データの前記記憶領域への配置の制約の大きさを示すアラインメントの降順であり、前記第2基準は、データの参照頻度の昇順であることとしてもよい。

これにより、所定サイズの記憶領域に無駄な空き領域を生じさせることなく、データの参照頻度を考慮した複数のデータの上記記憶領域内への配置が可能となるので、オブジェクトプログラムの実行効率を向上させることができる。

【図面の簡単な説明】

【図1】

本実施の形態におけるコンパイラ装置 1 0 0 の構成を示す。

【図 2】

ソースプログラム 2 0 5 の具体例を示す。

【図 3】

CPU 1 0 1 の行うスタック領域における変数の配置決定処理を示す。

【図 4】

変数対照表の具体例を示す。

【図 5】

変数 d のスタック領域における配置を示す配置模式図を示す。

【図 6】

変数 d, b のスタック領域における配置を示す配置模式図を示す。

【図 7】

変数 b, c, a のスタック領域における配置を示す配置模式図を示す。

【図 8】

変数 a, b, c, d のスタック領域における配置を示す配置模式図を示す。

【図 9】

配置決定処理によって決定されたスタック領域の配置に変数 a, b, c, d が配置された場合におけるオブジェクトプログラム例を示す。

【図 1 0】

従来法 1 によって変数 a, b, c, d がスタック領域に配置された場合におけるオブジェクトプログラム例を示す。

【図 1 1】

従来法 2 によって変数 a, b, c, d がスタック領域に配置された場合におけるオブジェクトプログラム例を示す。

【図 1 2】

従来法 1 における、コンパイラによるスタック領域における変数の配置の決定処理を示すフローチャートである。

【図 1 3】

従来法 1 によって決定されたスタック領域の配置に変数 a, b, c, d が配置された

場合における変数a,b,c,dのスタック領域における配置を示す配置模式図を示す。

【図14】

従来法2における、コンパイラによるスタック領域における変数の配置の決定処理を示すフローチャートである。

【図15】

従来法2によって決定されたスタック領域の配置に変数a,b,c,dが配置された場合における変数a,b,c,dのスタック領域における配置を示す配置模式図を示す。

【図16】

参照頻度に関する情報を含む変数対照表の具体例を示す。

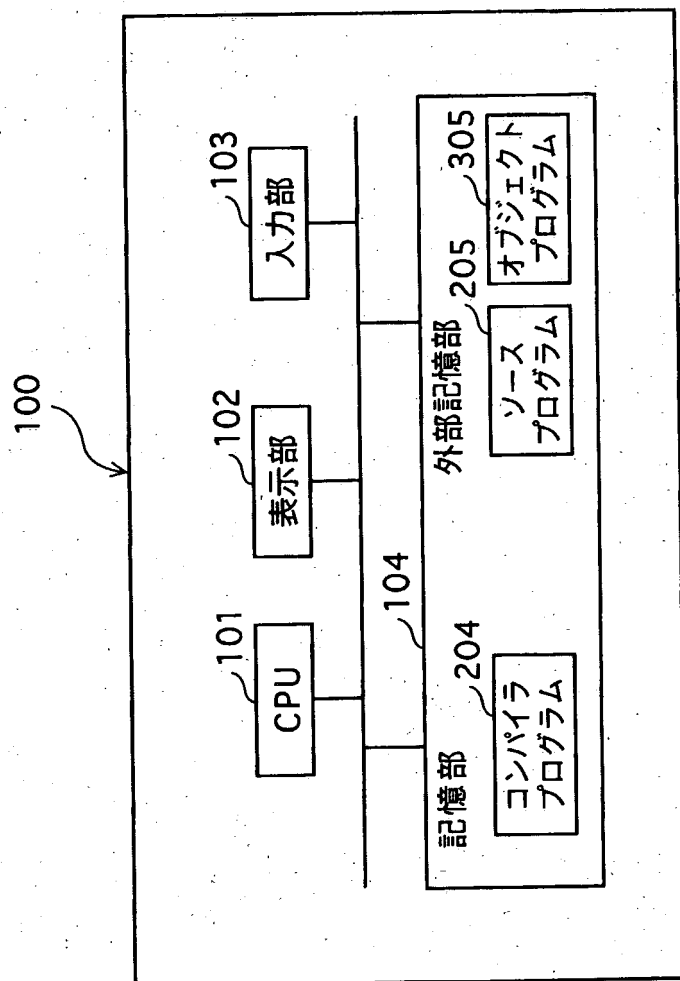
【符号の説明】

- 101 CPU
- 102 表示部
- 103 入力部
- 104 記憶部
- 204 コンパイラプログラム
- 205 ソースプログラム
- 305 オブジェクトプログラム

【書類名】

図面

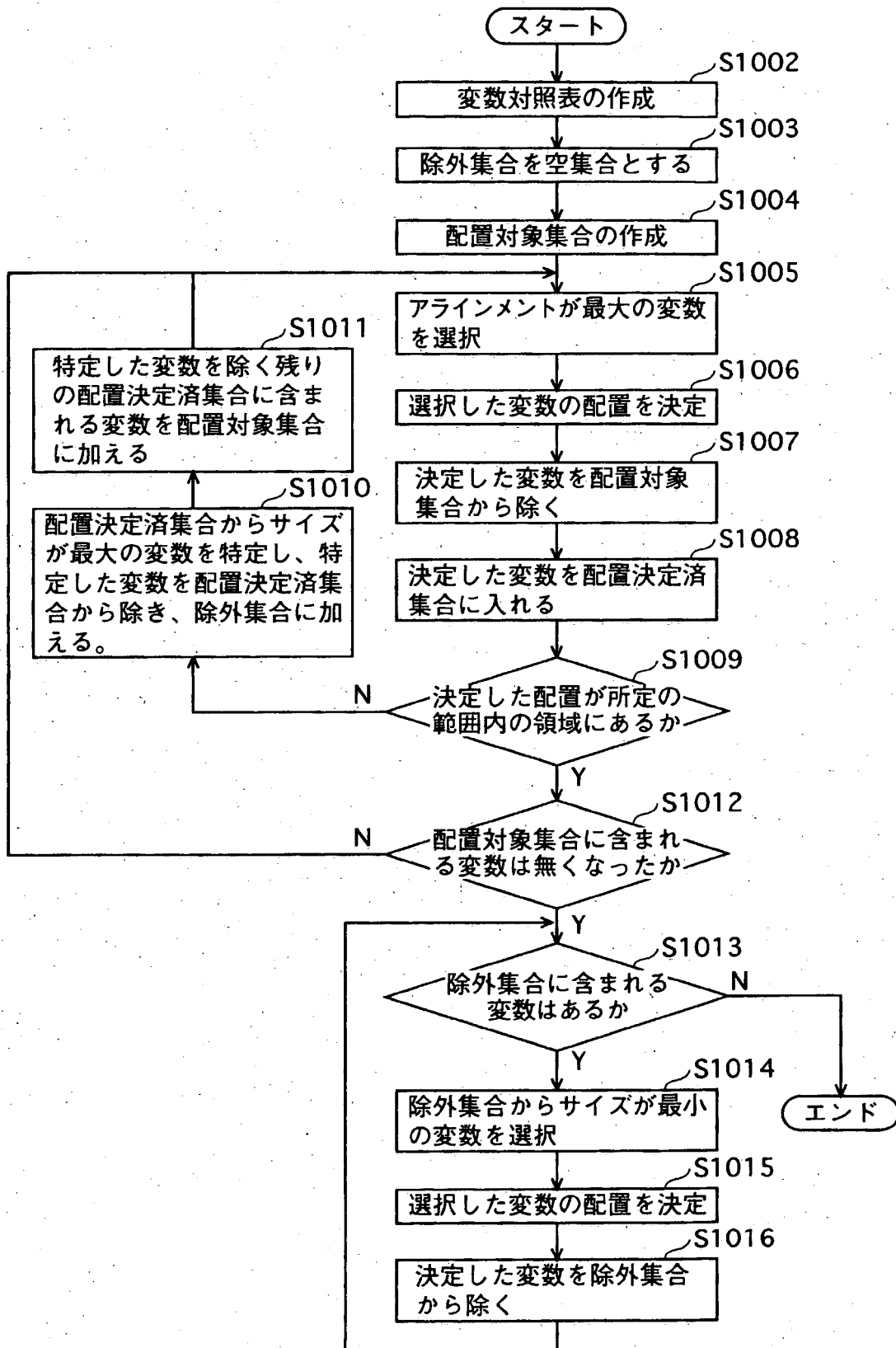
【図 1】



【図 2】

```
void dummy1(char);  
void dummy2(int);  
void dummy3(char*);  
void dummy4(double*);  
  
100 void  
101 f(void)  
102 {  
    .  
    .  
    .  
110 char a;  
111 int b;  
112 char c[19]  
113 double d[4]  
    .  
    .  
    .  
140 dummy1(a)  
141 dummy2(b)  
142 dummy3(c)  
143 dummy4(d)  
    .  
    .  
    .  
199 return  
200 }
```

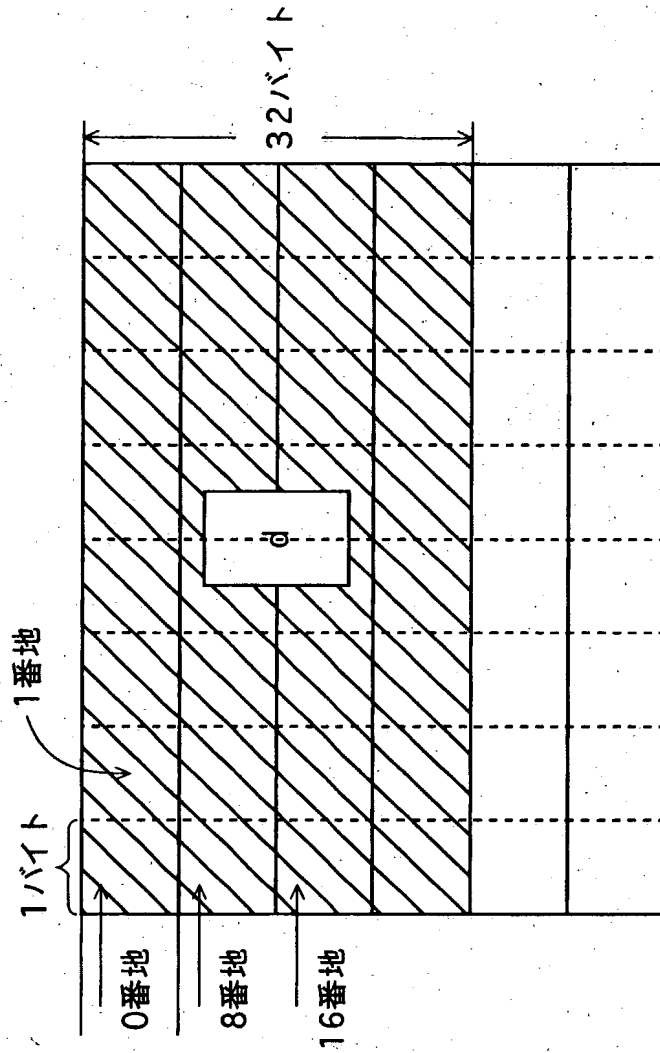
【図 3】



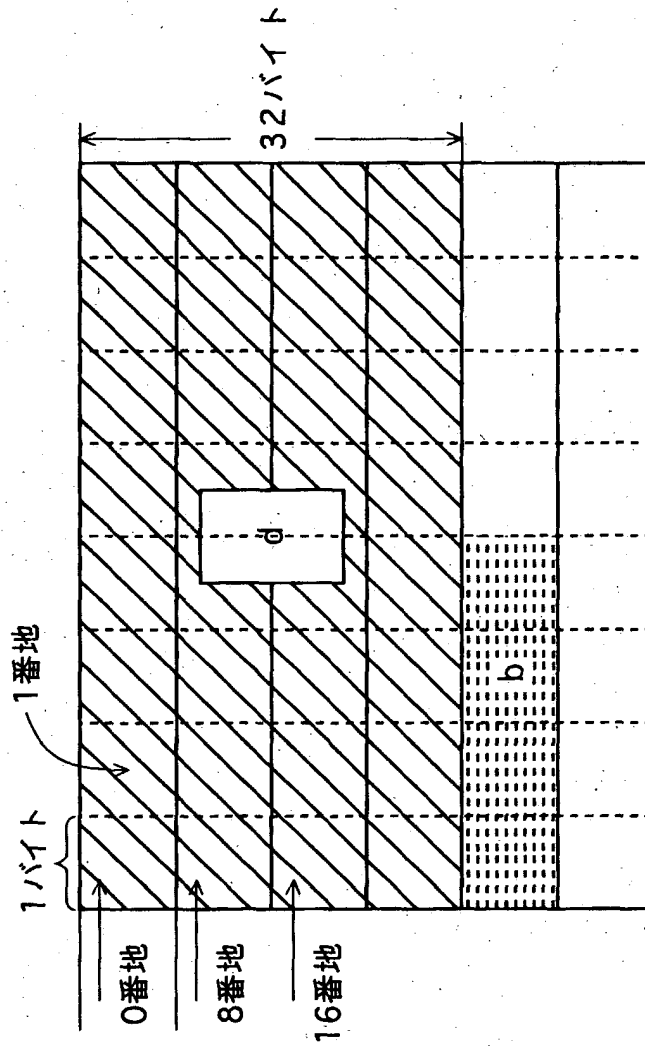
【図4】

型	変数名	データ サイズ	アライン メント
char	a	1	1
int	b	4	4
char[]	c	19	1
double[]	d	32	8

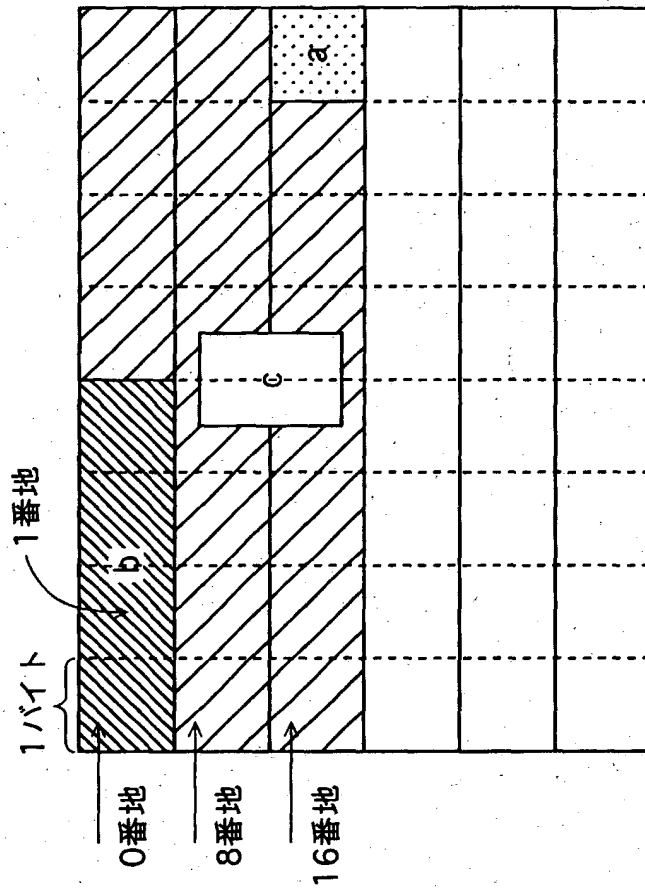
【図 5】



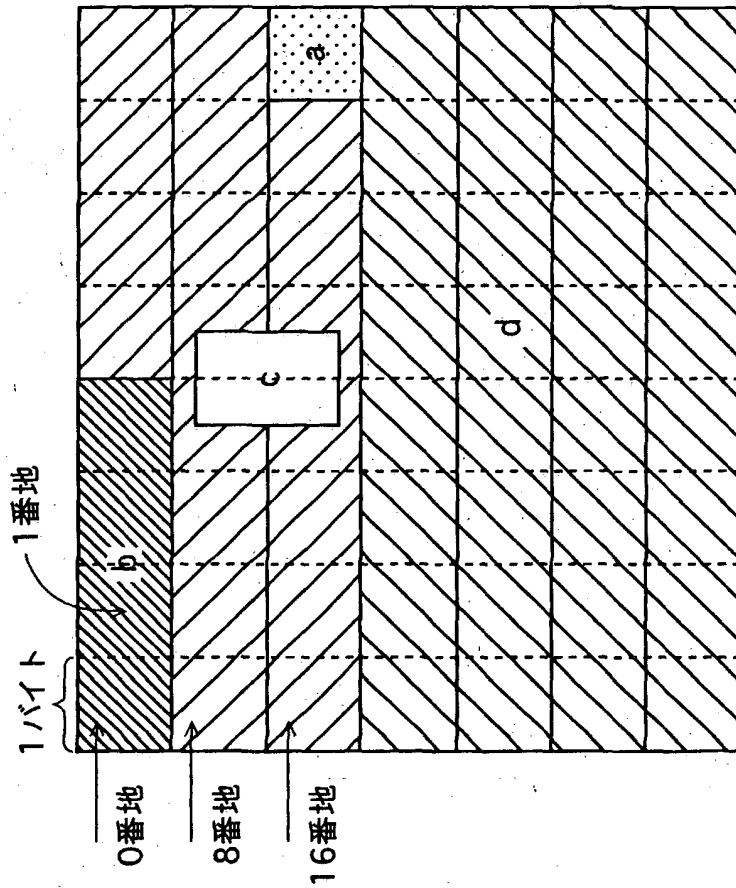
【図6】



【図7】



【図8】



【図 9】

210 ld1 r0,(sp,23) //aへのアクセス

・
・
・

220 ld4 r0,(sp,0) //bへのアクセス

・
・
・

230 ld1 r0,(sp,4) //cへのアクセス

・
・
・

240 ld8 r0,(sp,24) //dへのアクセス

・
・
・

【図 1 0】

310 ld1 r0,(sp,0) //aへのアクセス

・
・
・

320 ld4 r0,(sp,4) //bへのアクセス

・
・
・

330 ld1 r0,(sp,8) //cへのアクセス

・
・
・

340 mov r1,32 //dへのアクセス

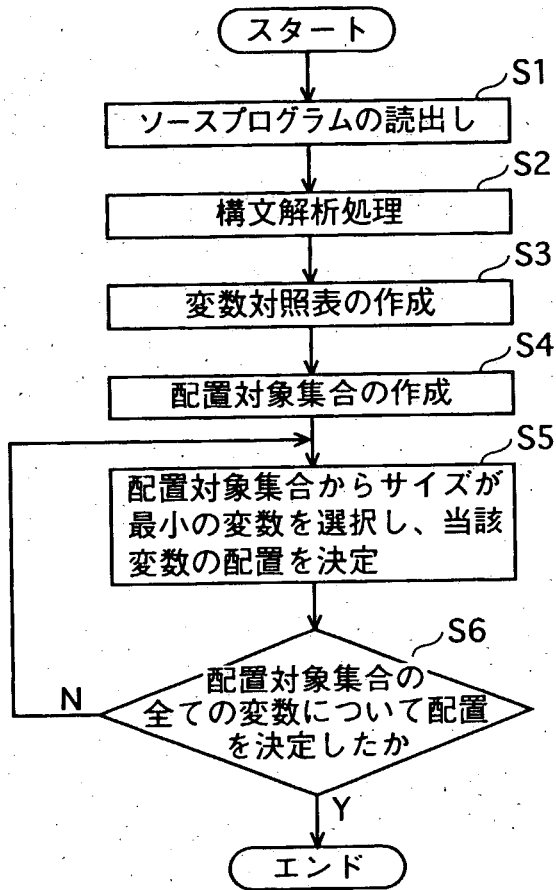
350 ld8 r0,(r1) //dへのアクセス

・
・
・

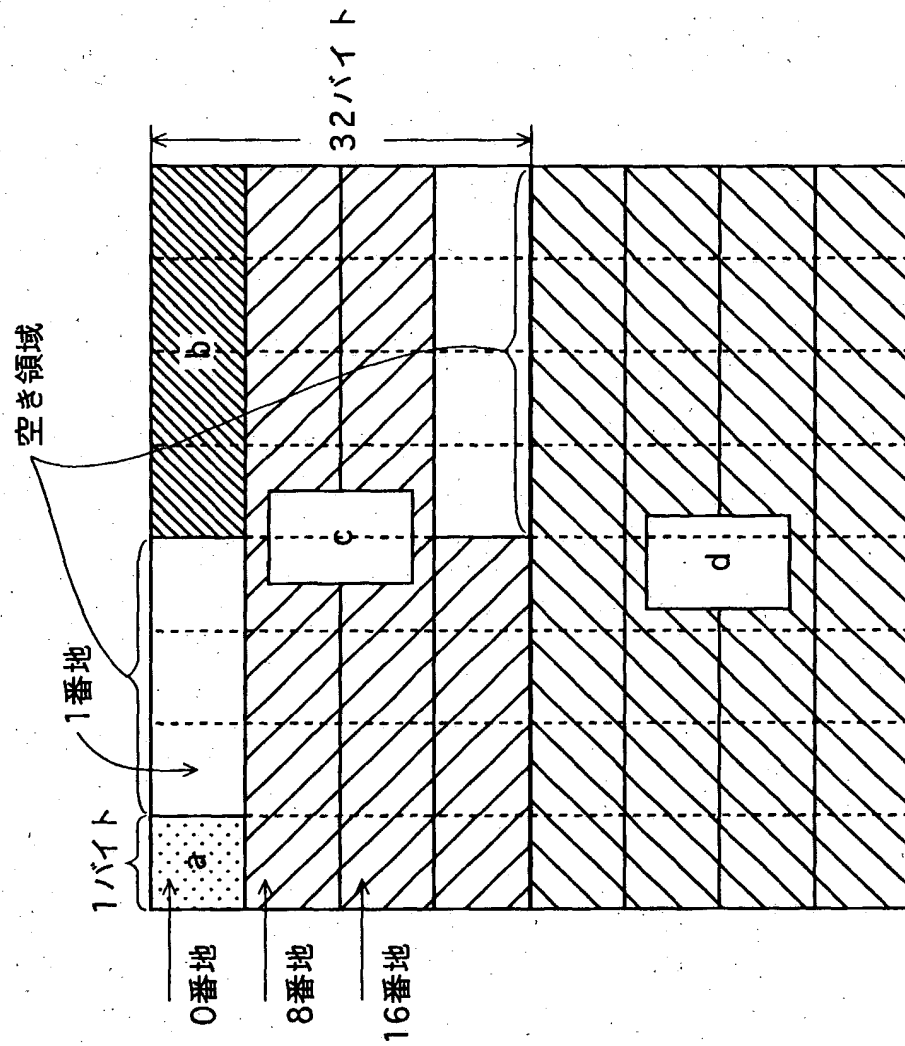
【図 1 1】

```
410      mov r1,55      //aへのアクセス
420      ld1 r0,(r1)     //aへのアクセス
      .
      .
      .
430      mov r1,32      //bへのアクセス
440      ld4 r0,(r1)     //bへのアクセス
      .
      .
      .
450      mov r1,36      //cへのアクセス
460      ld1 r0,(r1)     //cへのアクセス
      .
      .
      .
470      ld8 r0,(sp,0)   //dへのアクセス
      .
      .
      .
```

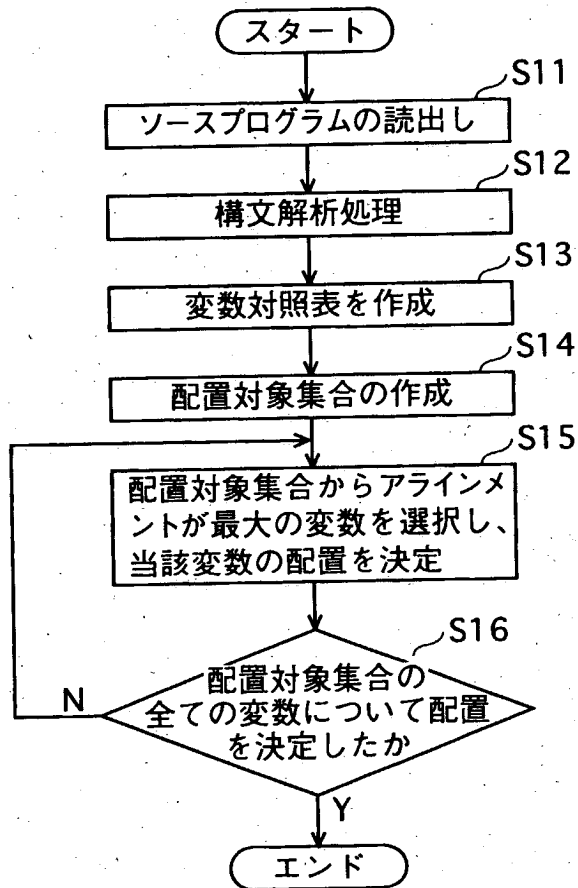
【図 12】



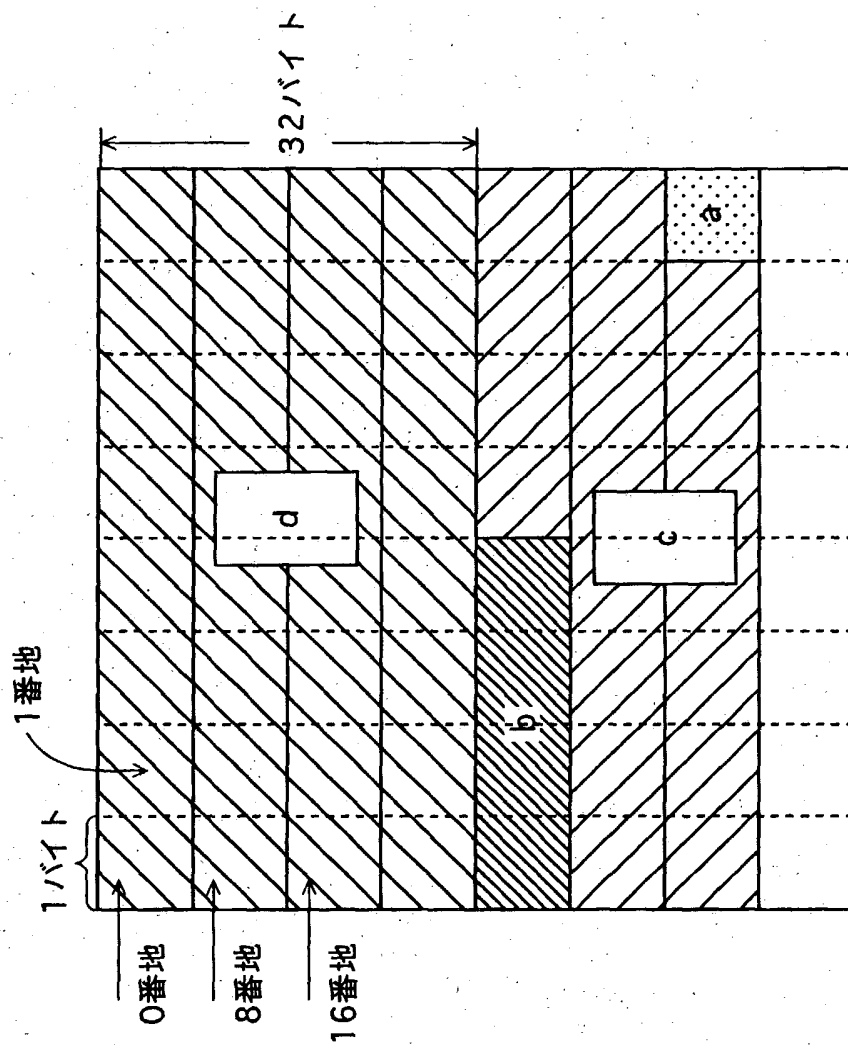
【図13】



【図 14】



【図15】



【図16】

型	変数名	サイズ	アライン	参照頻度
char	a	1	1	5
int	b	4	4	45
char[]	c	19	1	7
double[]	d	32	8	4

【書類名】 要約書

【要約】

【課題】 本発明は、スタック領域における変数の配置の決定を最適化するコンパイラ装置及び配置決定方法を提供することを目的とする。

【解決手段】 本発明は、複数のデータ属性を有する複数のデータの集合Xから、前記複数のデータ属性の何れかに基づいて定めた第1基準に従って、順次配置を決定すべきデータを選択し、データを選択する毎に、選択したデータの記憶領域内への配置の決定を試み、当該データの配置を決定できない場合毎に、選択したデータの中から除外対象となるデータを、前記第1基準に用いたデータ属性とは異なる第2基準に従って選択し、前記データ集合Xから前記除外対象データを除いたデータの中から、前記第1基準に従って、順次配置を決定すべきデータを選択する処理を、除外対象データを除く前記データ集合Xの全てのデータの配置が前記記憶領域内に入るように決定されるまで繰返すコンパイラ装置である。

【選択図】 図3

出 願 人 履 歴 情 報

識別番号 [000005821]

1. 変更年月日 1990年 8月28日

[変更理由] 新規登録

住 所 大阪府門真市大字門真1006番地

氏 名 松下電器産業株式会社